C246 Class Notes 2/25/03

Scribe: Claire Collins
Reader: Aaron Garnett

Administrative Notes:
-Office hours: Monday 5:30-7pm 461 Koshland or by appointment
-Exam #1: 3/4/03 in class, open book open note.

MULTIPLE ALIGNMENT:

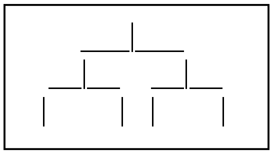
Q: Why do we do multiple alignments? What are the advantages of multiple alignments?

- see conserved residue positions
- make profiles that represent an entire family for alignment
- use to build trees
- see types of substitutions that occur (have a substitution matrix for each column of alignment

*multiple alignments are superior to pairwise alignments if they can be obtained.

PROGRESSIVE ALIGNMENTS:

- build a tree relating sequences – build guide trees using method: neighbor joining



1) do a series of pairwise alignments and get an alignment
2) do pairwise alignments between two alignments obtained in step 1 – profile/profile alignments

Feng-Doolittle:

Used Fitch-Margoliosh method for building a tree

- does all pairwise alignments
- turns similarity of alignments into distances
- plug distances into F/M algorithm to build the tree

CLUSTALW:

- progressive alignment
- ad hoc
- does all pairwise alignments
- build neighbor joining tree
- has to choose a matrix to use – you tell if you want it to be PAM or BLOSUM or Gonnet and it will try to guess which matrix to use based on first alignment it does.

*only used positive scores of matrix by default
*uses the basic idea of affine gaps by default

- calls GOP – gap opening penalty
- calls GEP – gap extension penalty

(Reference: Methods in Enzymology vol. 266)

GOP = (average mismatch value between 2 sequences)x(%identity)x
(original GOP + log (min(m,n)))

GEP = (original GEP)x(1 + |log(n/m)|)

GOP,GEP are also modified based on alignment around positions:

GOP = GOP(.3)(W/N)   W= # of seq. at that point without a gap
N = total # of sequences

The penalty for creating a gap decreases depending on how many sequences already have a gap at that point in the alignment.

GEP = GEP/2   *if aligned with gap of other sequence/alignment the GEP is less than normal

if making a gap where there is already a gap - less penalty

GOP = GOP (2 + (8-D2)/8)   D = distance to some other gap

*the penalty is increased if opening a gap near a region where gap is already open

Hydrophilic region (more likely to be on the surface, so more mutable):
GOP = 2(GOP)/3

Residue specific gap penalties: G (gap near glycine) .61
M (gap near methionine) 1.29
These are the penalties for if this residue occurs before the gap

CLUSTALW Other Features:

- end gaps not penalized
- "once a gap always a gap", the greatest flaw of progressive alignments
- not all information is available, so gap is at best place at that time, but it must stay there - when aligning this profile with others this gap stays, the end alignment will have this gap.

Improvements in CLUSTALW over CLUSTALV:

- sequence weighting (weight according to divergence)
- ability to add secondary structure information
- ability to align to a profile
- very laden with heuristics

Benefit of Progressive Alignment:

- fast
- evolutionarily logical

Disadvantages of Progressive Alignment:

- dependence on tree (which can't be a good tree because don't yet have optimal alignment)
- Numerous parameters
- ad hoc
- stuck in some sort of local minimum depending on how alignment created/tree constructed

TCoffee:

- uses CLUSTALW engine in beginning
- uses local alignment information
- tries to do pairwise alignment taking into account more information from tree
- each pairwise alignment is done through intermediaries

TCoffee (class paper) example:

A) GARFIELD THE LAST FAT CAT
B) GARFIELD THE FAST CAT
C) GARFIELD THE VERY FAST CAT
D) THE FAT CAT

when aligning A and B, C and D used as intermediates

The alignment without intermediates as guides:

```
A)   GARFIELD THE LAST FAT CAT
B)   GARFIELD THE FAST CAT ---
```

A and B aligned with C as intermediate

```
A)   GARFIELD THE LAST FA-T CAT
C)   GARFIELD THE VERY FAST CAT
B)   GARFIELD THE ---- FAST CAT
```

A and B aligned with D as intermediate

```
A)   GARFIELD THE LAST FA-T CAT
D)   -------- THE ---- FA-T CAT
B)   GARFIELD THE ---- FAST CAT
```

FINAL ALIGNMENT: with C and D used as intermediates

```
A)   GARFIELD THE LAST FA-T CAT
B)   GARFIELD THE ---- FAST CAT
```

*builds intermediates as master/slave to every other
sequence in alignment.
*still have once a gap always a gap
*fails less often than CLUSTALW

TCoffee gives better alignments than CLUSTALW, however:

- disadvantage: slower and more memory used

Iterative Algorithms:

* key idea: build up some initial alignment - use to guide future alignment
- MultAlign
-prrp (Goh - DP with affine in n squared time)
-DNR - doubly nested randomized iterative
1. starts with multiple alignments
2. make tree from multiple alignments
3. get pair weights from tree
4. optimize sum of pairs, tries bundling sequences based on similarity - pick various anchor points in alignment and realign between those points (used fixed anchor points)

- repeat

-IterAlign: tries to build alignment entirely iteratively (no tree used)

- Dialign: gives large number of small alignments and looks for smaller diagonals

- MAFFT (multiple alignment FFT)
  . what diagonals would align from 2 sequences using FFT
  . tells these diagonals have some interesting properties
  . fast
  . repeat for many pairs
  . does DP alignment algorithm to alignment
  . gives you offsets in 1-dimension from 1seq
  . can apply to multiple sequences